

CPSC 304 Project Cover Page

Milestone #: 4

Date: 12/01/2023

Group Number: 131

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Tara Ubovic	77903797	v8s8k	tarauboviccc@gmail.com
Madeline Paulson	36440824	d6o5e	madeline1paulson@gmail.com
Robin Matheson	80425994	z9p6u	robinm0211@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

Milestone 4

Project Description

Our project is a pet shelter management application. The project allows users to efficiently manage all areas of a pet shelter, including the adoption and care of animals, appointments, staff management, social media, event planning, and inventory management. This application utilizes the Oracle database management system to manage relationships between animals, adopters, employees and volunteers and facilitate a smooth and efficient adoption experience for everyone involved.

In the implementation, the app allows users to add, remove and update animals in the database. Our team's user-friendly interface simplifies adding new entries, removing outdated information, and updating existing details, ensuring the database remains accurate and comprehensive. The application provides valuable search capabilities, enabling users to locate a particular caretaker using criteria such as name, ID, or address. Additionally, users can identify all customers who have contributed donations surpassing a specified amount and determine the current count of animals belonging to a specific type within the pet shelter.

Schema Differences

We have had to make some changes to the schema and SQL table creations and insertions once we began implementing our project. We have had to combine each date and time attribute within the same relation since Oracle does not have distinct types for them, so it didn't make sense to keep them separated. We also changed the constraint syntax since we were getting errors when we wrote the create table statements as we were taught in class. Additionally, we were missing many ON DELETE CASCADE clauses.

- Every instance of VARCHAR changed to VARCHAR2() with a specified character limit as appropriate.
- Every relation with an ID attribute is now GENERATED ALWAYS AS IDENTITY.
- eventDayTime as a single attribute in FundraiserEvent as opposed to eventDate in Fundraiser and another relation FundraiserEventDate with eventDate and eventTime. eventDayTime is UNIQUE.
- vetDayTime as a single attribute in VetAppointment as opposed to vetDate and vetTime. vetDayTime now is the only attribute in the primary key.
- Adopter references Customer, added ON DELETE CASCADE.
- Animal now has arrivalDate instead of timeInShelter as it didn't make sense to have to regularly update every animal record. arrivalDate is NOT NULL.
- VetAppointment references Animal, added ON DELETE CASCADE.

- Worker references AnimalCaretaker, added ON DELETE CASCADE.
- Volunteer references AnimalCaretaker, added ON DELETE CASCADE.
- In Post added postingDate NOT NULL.
- AdoptionDetails references petID, added ON DELETE CASCADE.
- PetAdopter references Animal, added ON DELETE CASCADE.
- Appointment references Animal, added ON DELETE CASCADE.
- Appointment references Customer, added ON DELETE CASCADE.

Final Schema + Data Present

FundraiserEvent(**eventID**, eventType, eventDayTime, donationGoal);

AnimalCaretaker(**caretakerID**, caretakerName, fundEventID, caretakerAddress, caretakerPostalCode);

Customer(**customerID**, customerName);

Adopter(**adopterID**, numOfAdoptions, safeOwnerRating, adopterPostalCode, adopterAddress);

Animal(**petID**, animalName, type, age, favouriteCaretaker, previousOwner, arrivalDate, adopterID);

VetAppointment(**vetDayTime**, vetLicenseID, petID);

Worker(**workerID**, hourlyPay);

Volunteer(**volunteerID**, hoursVolunteered);

Post(**postID**, postType, description, postingDate, caretakerID);

AdoptionDetails(**adoptionID**, petID, adopterID, caretakerID, adoptionDate, notes);

PetAdopter(**petID**, adopterID);

Appointment(**petID**, **caretakerID**, **customerID**, apptDayTime);

Donation(**customerID**, **caretakerID**, amount);

Item(**itemID**, itemName, quantity);

ItemPurchase(**customerID**, **caretakerID**, **itemID**);

Vet(**vetLicenseID**, vetName);

AnimalCaretakerPC(**caretakerPostalCode**, caretakerCity);

PostDateAndType(**postingDate**, postType);

AdopterPC(**adopterPostalCode**, adopterCity);

ItemPrice(**itemID**, total);

FundraiserEvent

```
[SQL> select * from fundraiserevent;
```

EVENTID	EVENTTYPE	EVENTDAYT	DONATIONGOAL
1	Charity Auction	01-NOV-23	5000
2	Pet Walkathon	15-NOV-23	3000
3	Adoption Fair	30-NOV-23	2000
4	Pet Costume Contest	10-DEC-23	2500
5	Animal Rescue Gala	25-DEC-23	7000

AnimalCaretaker

CARETAKERID	CARETAKERNAME	FUNDEVENTID	CARETAKERADDRESS	CARETAKE
1	John Peters	1	123 Main St	12345
2	Mary Johnson	2	456 Elm St	67890
3	David Perks	3	789 Oak St	34567
4	Elaine Brown	4	101 Pine St	87654
5	Michael Wilson	5	234 Maple St	43210
6	Emily Anderson		123 Oak Lane	12345
7	Chris Martinez		789 Pine St	43210
8	Jasmine Walker		101 Maple St	34567
9	Ryan Turner		201 Cedar Drive	43210
10	Morgan Foster		234 Birch Blvd	67890

Customer

CUSTOMERID	CUSTOMERNAME
1	Alice Johnson
2	Bob Smith
3	Carol Davis
4	David Wilson
5	Eve Brown

Adopter

```
[SQL> select * from adopter;
```

ADOPTERID	NUMOFADOPTIONS	SAFEOWNERRATING	ADOPTERP	ADOPTERADDRESS
1	2	4	12345	123 Elm St
2	0	3	23456	456 Oak St
3	1	5	34567	789 Pine St
4	3	4	45678	101 Maple St
5	2	4	56789	234 Birch St

Animal

PETID	ANIMALNAME	TYPE	AGE	FAVOURITECARETAKER	PREVIOUSOWNER	ARRIVALDA	ADOPTERID
1	Fluffy	Cat	2	3	4	01-NOV-23	1
2	Rex	Dog	3	1	4	11-MAR-23	2
3	Whiskers	Cat	5	2	5	02-JAN-22	3
4	Buddy	Bunny	4	4	3	14-OCT-20	4
5	Luna	Dog	1	3	1	06-JUN-23	5
6	Domino	Hamster	1	3	1	23-FEB-22	
7	Patch	Dog	2	5	4	07-APR-23	
8	Pirate	Cat	2	4	4	11-MAY-21	
9	Cloudy	Bunny	3	2		11-NOV-23	
10	Smoothie	Bunny	3	2		31-OCT-23	

VetAppointment

```

[SQL> select * from vetappointment;

VETDAYTIM VETLICENSEID
-----
REASON
-----
      PETID
-----
20-OCT-23          1
Checkup
      1
21-OCT-23          2
Vaccination
      2
VETDAYTIM VETLICENSEID
-----
REASON
-----
      PETID
-----
22-OCT-23          3
Dental cleaning
      3
23-OCT-23          4
Spaying
VETDAYTIM VETLICENSEID
-----
REASON
-----
      PETID
-----
      4
24-OCT-23          5
Checkup
      5

```

Worker

```

[SQL> select * from worker;

  WORKERID  HOURLYPAY
-----
          1          15
          2          17
          3          14
          4          18
          5          16

```

Volunteer

```

[SQL> select * from volunteer;

VOLUNTEERID  HOURSVOLUNTEERED
-----
          6          20
          7          25
          8          18
          9          30
         10          22

```

Post

```
[SQL> select * from post;
```

POSTID	POSTTYPE	DESCRIPTION	POSTINGDA	CARETAKERID
1	Announcement	Adoption event this weekend!	25-OCT-23	1
2	News	New arrivals in the shelter	26-OCT-23	2
3	Update	Vet check-ups for all animals	27-OCT-23	3
4	Event	Volunteer appreciation day	28-OCT-23	3
5	Adoption	Adopt a furry friend today	29-OCT-23	1

AdoptionDetails

```
[SQL> select * from adoptiondetails;
```

ADOPTIONID	PETID	ADOPTERID	CARETAKERID	ADOPTIOND
-----	-----	-----	-----	-----
NOTES				

1	1	1	2	20-OCT-23
friendly cat				
2	2	2	2	21-OCT-23
playful dog				
3	3	3	3	22-OCT-23
loud cat				
ADOPTIONID	PETID	ADOPTERID	CARETAKERID	ADOPTIOND
-----	-----	-----	-----	-----
NOTES				

4	4	4	2	23-OCT-23
really soft bunny				
5	5	5	5	24-OCT-23
quiet dog				
6	6	5	2	25-OCT-23
interesting hamster				

PetAdopter

```
[SQL> select * from petadopter;
```

PETID	ADOPTERID
-----	-----
1	1
2	2
3	3
4	4
5	5
6	5

Appointment

```
[SQL> select * from appointment;
```

PETID	CARETAKERID	CUSTOMERID	APPTDAYTI
6	2	1	15-JAN-23
7	2	2	20-FEB-23
8	3	3	10-MAR-23
9	4	4	07-APR-23
10	5	5	12-MAY-23

Donation

```
[SQL> select * from donation;
```

CUSTOMERID	CARETAKERID	AMOUNT
1	1	100
2	2	150
3	3	200
4	4	50
5	5	75

Item

```
[SQL> select * from item;
```

ITEMID	ITEMNAME	QUANTITY
1	Pet Food	100
2	Blankets	50
3	Toys	75
4	Medicine	25
5	Leashes	30

ItemPurchase

```
[SQL> select * from itempurchase;
```

CUSTOMERID	CARETAKERID	ITEMID
1	1	3
2	1	5
3	3	1
4	5	4
5	3	5

Vet

```
[SQL> select * from vet;
```

VETLICENSEID	VETNAME
--------------	---------

1	Dr. Allan
2	Dr. Papper
3	Dr. Lorde
4	Dr. Levette
5	Dr. Michaels

AnimalCaretakerPC

```
[SQL> select * from animalcaretakerpc;
```

CARETAKE	CARETAKERCITY
----------	---------------

12345	Narnia
67890	Atlantis
34567	Brokeburn
87654	Lancaster
43210	Columbus

PostDateAndType

```
[SQL> select * from postdateandtype;
```

POSTINGDA	POSTTYPE
-----------	----------

25-OCT-23	Announcement
26-OCT-23	News
27-OCT-23	Update
28-OCT-23	Event
29-OCT-23	Adoption

AdopterPC

```
[SQL> select * from adopterpc;
```

ADOPTERP	ADOPTERCITY
----------	-------------

12345	Narnia
23456	Atlantis
34567	Brokeburn
45678	Lancaster
56789	Columbus

ItemPrice


```
[SQL> select * from itemprice;
```

ITEMID	TOTAL
5	200
2	100
1	150
4	75
3	90

SQL Queries + Locations

INSERT

Location: database_and_queries.php lines 132 to 212

DELETE

Location: database_and_queries.php lines 214 to 241

UPDATE

Location: database_and_queries.php lines 243 to 310

Selection

Location: database_and_queries.php lines 382 to 450

Projection

Location: database_and_queries.php lines 500 to 560

Join

Location: database_and_queries.php lines 464 to 498

Aggregation with Group By

Location: database_and_queries.php lines 563 to 593

Aggregation with Having

Location: database_and_queries.php lines 595 to 628

Nested Aggregation with Group By

Location: database_and_queries.php lines 630 to 660

Division

Location: database_and_queries.php lines 663 to 688

Query Screenshots

INSERT

Table from sqlplus before query:

```
SQL> SELECT * FROM ANIMAL;
```

PETID	ANIMALNAME	TYPE	AGE	FAVOURITECARETAKER	PREVIOUSOWNER	ARRIVALDA	ADOPTERID
1	Fluffy	Cat	2	3	4	01-NOV-23	1
2	Rex	Dog	3	1	4	11-MAR-23	2
3	Whiskers	Cat	5	2	5	02-JAN-22	3
4	Buddy	Bunny	4	4	3	14-OCT-20	4
5	Luna	Dog	1	3	1	06-JUN-23	5
6	Domino	Hamster	1	3	1	23-FEB-22	
7	Patch	Dog	2	5	4	07-APR-23	
8	Pirate	Cat	2	4	4	11-MAY-21	
9	Cloudy	Bunny	3	2		11-NOV-23	
10	Smoothie	Bunny	3	2		31-OCT-23	

10 rows selected.

Gui for the insert query:

Add an animal to the system

Animal Name:

Animal Type:

Age:

Favourite Caretaker ID:

Previous Owner ID:

Arrival Date:

Adopter ID:

Insert

Table after the query is executed:

Animal Table

Pet ID	Animal Name	Animal Type	Animal Age	Favourite Caretaker ID	Previous Owner	Arrival Date	Adopter ID
11	Petunia	bunny	3	5	2	30-NOV-23	
1	Fluffy	Cat	2	3	4	01-NOV-23	1
2	Rex	Dog	3	1	4	11-MAR-23	2
3	Whiskers	Cat	5	2	5	02-JAN-22	3
4	Buddy	Bunny	4	4	3	14-OCT-20	4
5	Luna	Dog	1	3	1	06-JUN-23	5
6	Domino	Hamster	1	3	1	23-FEB-22	
7	Patch	Dog	2	5	4	07-APR-23	
8	Pirate	Cat	2	4	4	11-MAY-21	
9	Cloudy	Bunny	3	2		11-NOV-23	
10	Smoothie	Bunny	3	2		31-OCT-23	

DELETE

Table from sqlplus before delete query is executed:

```
SQL> SELECT * FROM ANIMAL;
```

PETID	ANIMALNAME	TYPE	AGE	FAVOURITECARETAKER	PREVIOUSOWNER	ARRIVALDA	ADOPTERID
1	Fluffy	Cat	2	3	4	01-NOV-23	1
2	Rex	Dog	3	1	4	11-MAR-23	2
3	Whiskers	Cat	5	2	5	02-JAN-22	3
4	Buddy	Bunny	4	4	3	14-OCT-20	4
5	Luna	Dog	1	3	1	06-JUN-23	5
6	Domino	Hamster	1	3	1	23-FEB-22	
7	Patch	Dog	2	5	4	07-APR-23	
8	Pirate	Cat	2	4	4	11-MAY-21	
9	Cloudy	Bunny	3	2		11-NOV-23	
10	Smoothie	Bunny	3	2		31-OCT-23	

```
10 rows selected.
```

Delete query in the GUI:

Remove an animal from the system

Pet ID:

Delete

Table after the delete query is executed:

Animal Table

Pet ID	Animal Name	Animal Type	Animal Age	Favourite Caretaker ID	Previous Owner	Arrival Date	Adopter ID
1	Fluffy	Cat	2	3	4	01-NOV-23	1
2	Rex	Dog	3	1	4	11-MAR-23	2
3	Whiskers	Cat	5	2	5	02-JAN-22	3
4	Buddy	Bunny	4	4	3	14-OCT-20	4
5	Luna	Dog	1	3	1	06-JUN-23	5
7	Patch	Dog	2	5	4	07-APR-23	
8	Pirate	Cat	2	4	4	11-MAY-21	
9	Cloudy	Bunny	3	2		11-NOV-23	
10	Smoothie	Bunny	3	2		31-OCT-23	

UPDATE

Table from sqlplus before update query is executed:

```
SQL> SELECT * FROM ANIMAL;
```

PETID	ANIMALNAME	TYPE	AGE	FAVOURITECARETAKER	PREVIOUSOWNER	ARRIVALDA	ADOPTERID
1	Fluffy	Cat	2	3	4	01-NOV-23	1
2	Rex	Dog	3	1	4	11-MAR-23	2
3	Whiskers	Cat	5	2	5	02-JAN-22	3
4	Buddy	Bunny	4	4	3	14-OCT-20	4
5	Luna	Dog	1	3	1	06-JUN-23	5
6	Domino	Hamster	1	3	1	23-FEB-22	
7	Patch	Dog	2	5	4	07-APR-23	
8	Pirate	Cat	2	4	4	11-MAY-21	
9	Cloudy	Bunny	3	2		11-NOV-23	
10	Smoothie	Bunny	3	2		31-OCT-23	

10 rows selected.

Update query in the GUI:

Update animal information

The values are case sensitive and if you enter in the wrong case, the update statement will not do anything.

Pet ID:

Age:

Favourite Caretaker ID:

Adopter ID:

Update

Table after the update query is executed:

Animal Table

Pet ID	Animal Name	Animal Type	Animal Age	Favourite Caretaker ID	Previous Owner	Arrival Date	Adopter ID
1	Fluffy	Cat	18	5	4	01-NOV-23	4
2	Rex	Dog	3	1	4	11-MAR-23	2
3	Whiskers	Cat	5	2	5	02-JAN-22	3
4	Buddy	Bunny	4	4	3	14-OCT-20	4
5	Luna	Dog	1	3	1	06-JUN-23	5
6	Domino	Hamster	1	3	1	23-FEB-22	
7	Patch	Dog	2	5	4	07-APR-23	
8	Pirate	Cat	2	4	4	11-MAY-21	
9	Cloudy	Bunny	3	2		11-NOV-23	
10	Smoothie	Bunny	3	2		31-OCT-23	

Selection

Table from sqlplus before the selection query is executed:

```
SQL> SELECT * FROM ANIMALCARETAKER;
```

CARETAKERID	CARETAKERNAME	FUNDEVENTID	CARETAKERADDRESS	CARETAKE
1	John Peters	1	123 Main St	12345
2	Mary Johnson	2	456 Elm St	67890
3	David Perks	3	789 Oak St	34567
4	Elaine Brown	4	101 Pine St	87654
5	Michael Wilson	5	234 Maple St	43210
6	Emily Anderson		123 Oak Lane	12345
7	Chris Martinez		789 Pine St	43210
8	Jasmine Walker		101 Maple St	34567
9	Ryan Turner		201 Cedar Drive	43210
10	Morgan Foster		234 Birch Blvd	67890

```
10 rows selected.
```

Selection query in the GUI:

SELECTION: Find a Caretaker

Caretaker ID:

1

Caretaker Name:

John Peters

Fundraiser Event ID:

1

Caretaker Address:

123 Main St

Caretaker Postal Code:

12345

Submit

Results from the query:

Operation successful

Search Results

Caretaker ID	Caretaker Name	Fundraiser ID	Address	Postal Code
1	John Peters	1	123 Main St	12345

Projection

Table from sqlplus before the projection query is executed:

```
SQL> SELECT * FROM ANIMAL;
```

PETID	ANIMALNAME	TYPE	AGE	FAVOURITECARETAKER	PREVIOUSOWNER	ARRIVALDA	ADOPTERID
1	Fluffy	Cat	2	3	4	01-NOV-23	1
2	Rex	Dog	3	1	4	11-MAR-23	2
3	Whiskers	Cat	5	2	5	02-JAN-22	3
4	Buddy	Bunny	4	4	3	14-OCT-20	4
5	Luna	Dog	1	3	1	06-JUN-23	5
6	Domino	Hamster	1	3	1	23-FEB-22	
7	Patch	Dog	2	5	4	07-APR-23	
8	Pirate	Cat	2	4	4	11-MAY-21	
9	Cloudy	Bunny	3	2		11-NOV-23	
10	Smoothie	Bunny	3	2		31-OCT-23	

Projection query in the GUI:

PROJECTION: Choose which attributes you would like to see from Animals table

- ☒ petID
- ☒ animalName
- ☒ type
- ☐ age
- ☐ Animal's Favorite Caretaker
- ☐ Animal's Previous Owner
- ☒ Animal's Arrival Date to the Shelter
- ☐ Adopter ID

submit

Results of the projection query:

Operation successful

Search Results

petID	animalName	type	arrivalDate
1	Fluffy	Cat	01-NOV-23
2	Rex	Dog	11-MAR-23
3	Whiskers	Cat	02-JAN-22
4	Buddy	Bunny	14-OCT-20
5	Luna	Dog	06-JUN-23
6	Domino	Hamster	23-FEB-22
7	Patch	Dog	07-APR-23
8	Pirate	Cat	11-MAY-21
9	Cloudy	Bunny	11-NOV-23
10	Smoothie	Bunny	31-OCT-23

Join

Tables from sqlplus before the join query is executed:

```
SQL> SELECT * FROM CUSTOMER;
```

CUSTOMERID	CUSTOMERNAME
1	Alice Johnson
2	Bob Smith
3	Carol Davis
4	David Wilson
5	Eve Brown

```
SQL> SELECT * FROM DONATION;
```

CUSTOMERID	CARETAKERID	AMOUNT
1	1	100
2	2	150
3	3	200
4	4	50
5	5	75

Join query in the GUI:

JOIN: Find customers who made donations above a certain amount

Donation Amount:

Submit

Results of the join query:

Operation successful

Search Results

Customers with Donations above 80

Customer Name	Donation Amount
Alice Johnson	100
Bob Smith	150
Carol Davis	200

Aggregation with Group By

Table from sqlplus before the group by query is executed:

```
SQL> SELECT * FROM ANIMAL;
```

PETID	ANIMALNAME	TYPE	AGE	FAVOURITECARETAKER	PREVIOUSOWNER	ARRIVALDA	ADOPTERID
1	Fluffy	Cat	2	3	4	01-NOV-23	1
2	Rex	Dog	3	1	4	11-MAR-23	2
3	Whiskers	Cat	5	2	5	02-JAN-22	3
4	Buddy	Bunny	4	4	3	14-OCT-20	4
5	Luna	Dog	1	3	1	06-JUN-23	5
6	Domino	Hamster	1	3	1	23-FEB-22	
7	Patch	Dog	2	5	4	07-APR-23	
8	Pirate	Cat	2	4	4	11-MAY-21	
9	Cloudy	Bunny	3	2		11-NOV-23	
10	Smoothie	Bunny	3	2		31-OCT-23	

Aggregation with group by query in the GUI:

Aggregation with GROUP BY: Count how many animals of specific type we have in pet shelter

The values are case sensitive and if you enter in the wrong case, the group by query will not be correctly executed.

Reminder: Animal type first letter should be capitalized.

Type of animal:

Submit

Results of the query:

Operation successful

Search Results

Animal Type	TypeCount
Cat	3

Aggregation with Having

Table in sqlplus before the having query is executed:

```
SQL> SELECT * FROM FUNDRAISEREVENT;
```

EVENTID	EVENTTYPE	EVENTDAYT	DONATIONGOAL
1	Charity Auction	01-NOV-23	5000
2	Pet Walkathon	15-NOV-23	3000
3	Adoption Fair	30-NOV-23	2000
4	Pet Costume Contest	10-DEC-23	2500
5	Animal Rescue Gala	25-DEC-23	7000

Aggregation with having query in the GUI:

Aggregation with HAVING: Find fundraiser event types with specified average donation goal or above

Average Donation Goal Value:

Submit

Results of the query:

Operation successful

Search Results

Event Type	Average Donation Amount
Pet Walkathon	3000
Charity Auction	5000
Animal Rescue Gala	7000

Nested Aggregation with Group By

Tables before the nested aggregation with group by query is executed:

```
SQL> SELECT * FROM CUSTOMER;
```

CUSTOMERID	CUSTOMERNAME
------------	--------------

1	Alice Johnson
2	Bob Smith
3	Carol Davis
4	David Wilson
5	Eve Brown

```
SQL> SELECT * FROM ITEMPURCHASE;
```

CUSTOMERID	CARETAKERID	ITEMID
------------	-------------	--------

1	1	3
2	1	5
3	3	1
4	5	4
5	3	5

Nested aggregation with group by query in the GUI:

Nested Aggregation with GROUP BY: Lists customers that have purchased same or more number of items than the average customers

Submit

Results of the query:

Operation successful

Search Results

Customer ID	Customer Name	Average Number of Items purchased per Customer
3	Carol Davis	2
2	Bob Smith	2
4	David Wilson	2
1	Alice Johnson	2

Division

Tables before the division query is executed:

```
SQL> SELECT * FROM ANIMAL;
```

PETID	ANIMALNAME	TYPE	AGE	FAVOURITECARETAKER	PREVIOUSOWNER	ARRIVALDA	ADOPTERID
1	Fluffy	Cat	2	3	4	01-NOV-23	1
2	Rex	Dog	3	1	4	11-MAR-23	2
3	Whiskers	Cat	5	2	5	02-JAN-22	3
4	Buddy	Bunny	4	4	3	14-OCT-20	4
5	Luna	Dog	1	3	1	06-JUN-23	5
6	Domino	Hamster	1	3	1	23-FEB-22	
7	Patch	Dog	2	5	4	07-APR-23	
8	Pirate	Cat	2	4	4	11-MAY-21	
9	Cloudy	Bunny	3	2		11-NOV-23	
10	Smoothie	Bunny	3	2		31-OCT-23	

```
SQL> SELECT * FROM ANIMALCARETAKER;
```

CARETAKERID	CARETAKERNAME	FUNDEVENTID	CARETAKERADDRESS	CARETAKE
1	John Peters	1	123 Main St	12345
2	Mary Johnson	2	456 Elm St	67890
3	David Perks	3	789 Oak St	34567
4	Elaine Brown	4	101 Pine St	87654
5	Michael Wilson	5	234 Maple St	43210
6	Emily Anderson		123 Oak Lane	12345
7	Chris Martinez		789 Pine St	43210
8	Jasmine Walker		101 Maple St	34567
9	Ryan Turner		201 Cedar Drive	43210
10	Morgan Foster		234 Birch Blvd	67890

10 rows selected.

```
SQL> SELECT * FROM ADOPTIONDETAILS;
```

ADOPTIONID	PETID	ADOPTERID	CARETAKERID	ADOPTIOND	NOTES
1	1	1	2	20-OCT-23	friendly cat
2	2	2	2	21-OCT-23	playful dog
3	3	3	3	22-OCT-23	loud cat
4	4	4	2	23-OCT-23	really soft bunny
5	5	5	5	24-OCT-23	quiet dog
6	6	5	2	25-OCT-23	interesting hamster

6 rows selected.

Division query in the GUI & the result of the query:

DIVISION: Caretakers Facilitating Adoption of Every Animal Type

Submit

Operation successful

Search Results

Caretaker ID	Caretaker Name
2	Mary Johnson